

## **AMENDMENTS TO THE SPECIFICATION**

On page 1, after the inventor name, please add new section:

### **RELATED APPLICATION**

**[0000]** This application is related to U.S. application serial number 10/061,548, filed on January 31, 2002 by David J. Hudson, titled "Netbufs: Communication Protocol Packet Buffering Using Paged Memory Management."

Please amend the following paragraphs as indicated below:

**[0005]** The Berkeley Unix system implemented a strategy known as mbufs, each of which contains a portion of the data within a communication packet. The mbufs are joined together via various forms of linked lists, which are in turn iterated around by special helper software functions to provide an apparently contiguous view of the packet to their users. The designs of the mbufs are not suited to the present problem however because they assume that any information about the individual mbufs may be stored within the mbufs themselves and thus the run-time cost of accessing individual elements within them is quite high. In addition the mbufs are stored within the main system memory, that is impractical in limited memory devices. Finally, while mbufs offer a clustering solution that allows the sharing of data between two or more chains of buffers, the sharing introduces another level of indirection and run-time cost.

**[0006]** Linux utilizes a system called of skbuffs, each of which is a relatively large fixed-size memory area and which may be grouped together if necessary to form larger blocks. The larger size allows for efficient run-time operation, however the size of a single skbuff may be greater than the memory available within some embedded systems to which the

invention may be applied, and multiple skbuffs must be available in order to implement any useful form of communications. Skbuffs also suffer the same problems as mbufs with their requirement to use main system memory.

[0010]        A communication protocol packet buffering system and method using paged memory management that provides a mechanism for a microprocessor-based system to manage communication (network) data packets is provided according to the present invention. This buffering management strategy includes the individual network packet buffers (netbufs). Netbufs provide a general-purpose mechanism for managing communication packets and are not tied to any particular type of communication protocol. As such it is possible for the microprocessor-based system to use different netbufs for different purposes at the same time. For example, netbufs can be used in situations where one or more communications protocols are in use, such as in protocol converters or gateways. The netbufs allow a general-purpose implementation of a range of communication protocols within a microprocessor-based system that is conventionally deemed to have a memory capacity that is too small. Customers are thus able to implement applications in situations that would either have been previously impractical for reasons of cost or physical integration, or alternatively with lower-cost hardware.

[0036]        The fixed size of the netpages makes the overhead associated with allocation very low as it may be handled with a simple free-list. Netpages may be either allocated or free. If the netpages they are allocated then they are in use by one or more netpages netbufs, whereas if they are free then no netbufs reference them. Associated with the physical netpages is a table containing reference count information or pointers to the list of unused, or free, netpages. When a netpage is in use the reference count identifies how many netbufs index

that specific netpage. When the page is not in use, the table entry contains a pointer to the next unused page, if any, to form a list of unused netpages.

[0047] For example, initially an empty netbuf (containing no data) will have all pointers referencing the middle of the logical address space. Subsequent data write operations may occur with either increasing or decreasing logical addresses allowing a buffer that is written with entirely decreasing or entirely increasing logical addresses or a combination of both ~~the~~ to be accommodated within the available netpage data storage space that is indexed by the netbuf.